The Virtual Reality Responsive Workbench: Applications and Experiences

Lawrence Rosenblum, James Durbin, Robert Doyle, and David Tate
Virtual Reality Lab/Information Technology Division
Naval Research Laboratory
Washington, DC 20375

1.0 Introduction

Virtual Reality (VR) is a complex and challenging field [Earnshaw and Rosenblum, 1995; Rosenblum and Cross, 1997], and several distinct types of systems have been developed for displaying and interacting with virtual environments. One of the newest is the Virtual Reality Responsive Workbench [Kreuger and Froehlich, 1994; Kreuger et al., 1995; Rosenblum, Bryson, and Feiner, 1995]. The Workbench is an interactive VR environment designed to support a team of end users such as military and civilian command and control specialists, designers, engineers, and doctors. The Virtual Workbench creates a match for the "real" work environment of persons who would typically stand over a table or a workbench as part of their professional routine. For example, the Workbench could be used to represent fluid flow over a ship's hull while supporting a design team in interactive visualization. Perhaps the greatest strength of the VR Responsive Workbench is the ease of natural interaction with virtual objects. Current interactive methods emphasize gesture recognition, speech recognition, and a simulated "laser" pointer to identify and manipulate objects.

This paper classifies VR systems into three categories: immersive head-mounted displays (HMDs), immersive non-HMD systems, and partially immersive tabletop systems. We discuss the utility of each classification. Several applications that we have developed in the Virtual Reality Laboratory of the Information Technology Division (ITD), Naval Research Laboratory (NRL) are examined, and we discuss our experiences with VR Responsive Workbench interfaces and software architecture.

2.0 Systems for VR

There is no accepted definition for VR. One important reference, the U.S. National Research Council report *Virtual Reality: Scientific and Technical Challenges* [Durlach and Mavor, 1995], does not attempt a definition. Rather, characteristics of a virtual environment are given. These include a man-machine interface between human and computer, 3D objects, objects having a spatial presence independent of the user's position, and the user manipulating objects using a variety of motor channels. Virtual reality can be subdivided in many different ways; here we will categorize based upon the visual channel.

2.1 Head-Mounted Displays/BOOMs:

Head-mounted displays (HMDs), which typically also include earphones for the auditory channel as well as devices for measuring the position and orientation of the user, have been the primary VR visual device for much of the 1990's. Using CRT or LCD technology, HMDs provide two imaging screens, one for each eye. Thus, given sufficient computer power, stereographic images are generated. Typically, the user is completely immersed in the scene, although HMDs for augmented

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **APR 1997**	2. REPORT TYPE	3. DATES COVERED **00-00-1997 to 00-00-1997**

4. TITLE AND SUBTITLE **The Virtual Reality Responsive Workbench: Applications and Experiences**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Naval Research Laboratory,4555 Overlook Ave. SW,Washington,DC,20375**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
British Computer Society Conference on Virtual Worlds on the WWW, Internet, and Networks, Bradford, UK, April 1997.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**13**	

reality overlay the computer-generated image onto the real-world at low resolutions. Low-end HMDs can be obtained for less than $10,000. These suffer from information loss (resolutions of approximately 400 x 300 pixels; typical field-of views between 40 deg. to 75 deg.). Extremely low end "glasses" cost only hundreds of dollars, but these systems are not yet usable for serious applications and find their role in system testing and in research. High-end HMDs overcome these limitations at very high costs and thus are utilized only for a limited number of applications such as military flight training. In addition, ergonomic limitations such as weight, fit, and isolation from the real environment make it unlikely that users will accept HMD-based immersion for more than short time periods until such time as advances in material science produce eyeglass size and weight HMDs. They are, however, more portable than are other VR systems.

An alternative to HMDs is the BOOM (Binocular Omni-Orientation Monitor). Two high-resolution CRTs are mounted inside a package against which the user places his eyes. By counterbalancing the CRT packaging on a free-standing platform, the display unit allows the user six-degree-of-freedom movement while placing no weight on the user's head. The original version of the BOOM had the user navigating through the virtual world by grasping and moving two handles and turning the head display much as one would manipulate a pair of binoculars. Buttons on the hand-grip are available for user input. A more recent desktop version (the Fakespace PushBOOM) allows the user to navigate by pushing his head against a spring-loaded system.

HMDs and BOOMs are similar devices in that the user is fully immersed in the virtual environment and does not see his actual surroundings. The BOOM solves several of the limitations of the HMD (e.g., resolution, weight, field-of-view), but at the expense of reducing the sense of immersion by requiring the user to stand or sit in a fixed position. This loses the freedom of movement associated with HMDs where users typically take steps and turn their body to determine direction (the BOOM also restricts the user's hands).

2.2 Immersive Rooms:

Immersion does not necessarily require the use of the head-mounted displays that are the most common method for presenting the visual channel in a virtual environment. The CAVE™ (CAVE Automatic Virtual Environment), a type of Immersive Room facility developed at the University of Illinois, Chicago, accomplishes immersion by projecting on two or three walls and a floor and allowing the user to interactively explore a virtual environment [Cruz-Neira et al., 1993]. An Immersive Room is typically about 10' by 10' by 13' (height), allowing a half-dozen or more users to examine the virtual world being generated within the space. Computer-generated stereographic images are produced by calculating right and left eye images and using stereographic shuttered glasses to synchronize these alternating images at 120 Hz. To determine the view, a single group leader is head tracked using magnetic sensors to determine position and orientation. Both by walking within the Immersive Room and by utilizing an interactive device called a "wand," which has a second tracker for position identification and buttons for issuing commands, the group leader navigates through the data. All users see the same image; thus, other team members view the scene from an incorrect perspective with the resulting distortion depending upon differences in location within the Immersive Room. Since the stereographic shuttered glasses are see-through, all users see each other. This facilitates group discussion and data analysis.

While HMDs require that users interact in virtual spaces (they cannot see each other in their "real" environment), the Immersive Room offers the significant advantage of permitting user interaction, discussion, and analysis in the real world. However, the computational cost of generating scenes

within an Immersive Room are very high. Two images must be generated at high refresh rates for each wall in the Immersive Room. In addition, each wall requires a high-quality projector and, since back projection is used, a large allocation of space is required for projection length. Costing over one-half million dollars, Immersive Rooms exist only in a handful of large research organizations and corporations.

2.3 The VR Responsive Workbench:

The two paradigms discussed above are both fully immersive. However, there are many applications for which full immersion is not desirable. A doctor performing pre-surgical planning has no reason to wish to be fully immersed in a virtual room and with virtual equipment. Rather, he would like a virtual patient lying on an operating table in a real room. He would like to reach out and interactively examine the virtual patient and, perhaps, practice the operation. Similar remarks apply to engineering design, military and civilian command and control, architectural layout, and a host of other applications that would typically be performed on a desktop, table, or workbench. These applications are categorized by not requiring navigating through complex virtual environments but rather by demanding a fine-granularity visualization and interaction with virtual objects and scenes. Thus, the Workbench supports VR for a large class of applications that are substantially different from the fully immersed, navigation-oriented applications supported by HMDs and Immersive Rooms.

The Virtual Workbench operates by projecting a computer-generated, stereoscopic image off a mirror and then onto a table (i.e., workbench) surface that is viewed by a group of users around the table (e.g., Figure 1). Using stereoscopic shuttered glasses (just as is done in the Immersive Room™, users observe a 3D image displayed above the tabletop. By tracking the group leader's head and hand movements using magnetic sensors, the Virtual Workbench permits changing the view angle and interacting with the 3D scene. Other group members observe the scene as manipulated by the group leader, facilitating easy communication between observers about the scene and defining future actions by the group leader. Interaction is performed using speech recognition, a pinch glove for gesture recognition, and a simulated laser pointer. Figure 1 shows a schematic of the Workbench.

Table 1 presents trade-offs between these systems. Table 2 indicates the strengths of each type of system.

	HMD (mid-range)	PushBOOM	Immersive Room	Workbench
Immersion	Full	Full	Full	Partial
Resolution	Low	High	Medium	Medium
Habitability[3]	Poor	Fair	Fair/Good	Good
Detailed Interaction	Low	Low	Low/Medium	High
Group Interactions	Low	Low	High	High
Portability	High	High	None	Low
Cost (device only)	$10K[1]	$35K[1]	$150K[2]	$60K[1]

Table 1 - System Characteristics

[1] Requires high-end graphics workstation for most applications, cost approx. $150,000; some applications can be performed using less expensive computational engines
[2] Requires multi-pipe, high-end graphics workstation, cost approx. $400,000.
[3] Refers to the willingness of users to stay within the virtual environment

	HMD/BOOM	Immersive Room	Workbench
Strengths	Navigation	Navigation Collaboration	Detailed Interaction Collaboration
Sample Applications	Architecture Walk-through Single-User Mission Rehearsal	Scientific Visualization Information Visualization Multi-User Mission Rehearsal Engineering Design	Medicine Engineering Design Mission Planning Scientific Visualization Data Mining

Table 2 - System Usage

In 1994, the NRL/ITD VR Lab designed and fabricated the first Virtual Reality Responsive Workbench in the U.S. [Rosenblum, Bryson, and Feiner, 1995] based in part upon earlier work at the German National Research Center for Information Sciences (GMD). The remainder of this paper discusses our interactive methods for the Workbench as well as two applications: medicine and situational awareness. A third application that we have developed, not discussed in this paper, is engineering design where we show how the Workbench was used to find new information about a preliminary version of a ship design [Rosenblum et al., 1996].

3.0 Interactive Techniques

This section discusses three types of interactive techniques that we employ on the Workbench: (1) direct manipulation using a "pinch glove"-like system, (2) voice recognition, and (3) a simulated laser pointer ("wand").

3.1 Direct Manipulation via a Glove:

For direct manipulation, a user places an instrumented hand into the virtual environment and attempts to interact with virtual objects as if they were physically present. For example, in our medical application a user can reach into the skeleton and select and grab bone groups or internal

organs. Grabbing is accomplished via a pinch-glove-like system. The user makes a pinching gesture with his index finger and thumb to indicate a desire to grab the currently selected item. To let the user know that a specific object is selectable, we change the color of the object when the user's hand in close to the object. Once grabbed, the object is attached to the hand. The user manipulates it as they would a real object; in/out and sideways hand movements zoom and pan while hand rotations rotate the object.

The strength of this metaphor is that it is simple and intuitive. The user interacts exactly as he would with a real-life object, reaching out, grabbing it, and manipulating it. There is no artificiality and the need for a learning process for the interaction ("mouse-ology") is eliminated. We have found that users adapt to the metaphor in seconds. It largely fulfills the goal of VR, producing a natural environment where objects have "presence" (i.e., they are treated and act exactly as real objects would).

While this method is natural and intuitive, issues remain to be solved. The most obvious is that the user must physically wear the glove and that the glove is attached by wires to a control box. Ideally one would like to recognize gestures without using a glove. Investigations into natural gesture recognition are taking place using techniques such as optical flow to identify hand motion. However, better algorithms and faster computers are required before the glove can be replace by a camera-based system. Our glove metaphors are limited to natural actions such as grabbing and touching. Glove-based systems that require users to learn, memorize, and remember unnatural and non-intuitive gesture combinations defeat the purpose of VR.

Another issue in glove-based gesture recognition is that the instrumented hand blocks some of the imagery. To perceive stereo images correctly on the Workbench, the user's visual system focuses on the imaging surface (table top). However the eyes also must converge at a point in space necessary to correctly perceive the depth of a specific object. For example, the eyes converge above the table for images intended to be above the imaging surface and converge below the table for images intended to be behind the imaging surface. This is not how the human visual system usually works and thus takes a user some time to adjust. Eye strain can result. Introducing the user's hand into the projected space causes the eyes to attempt to both focus on imaging surface and converge elsewhere and focus and converge on real physical object. Usually the human's visual system will default to what is normal: it will focus and converge on the real object and lose the correct perception of the projected object. This often causes a user difficulty in selecting objects as they lose their depth cues the closer the real world hand comes to the projected virtual object.

NRL has also developed a two-handed glove system [Obeysekare et al., 1996]. In this system the user wears both a right-handed and left-handed pinch glove and can pick up objects in each hand. The two-handed glove has been used to examine molecular manipulation and related applications.

3.2 Speech Recognition:

The ability to issue verbal commands to a computer and have the computer understand and respond has long been a desired goal of the human-computer interface community. Ideally, the computer would understand conversational English, including the correct handling of pronouns, context between sentences, and continuous speech. Researchers are beginning to produce systems that move toward this goal. However, systems available today have less capability.

Speech input is ideal because speech is natural. A typical person interacts with many other humans daily and uses speech as the primary communication channel. Many humans even talk to inanimate objects such as their cars and toaster ovens, although those inanimate objects have no speech recognition capacity whatsoever. Humans are simply comfortable with speaking to or at things in an attempt to communicate.

A well-designed system will support common commands that a user might want to execute. These commands are often highly dependent on the task that the application was designed to solve. The exact formation of these commands will either be flexible (i.e. multiple ways to state intended interactions) and/or very obvious to a typical user of the system. Verbal commands are often much easier for a human to remember than a contrived keyboard key combination or button combination on an input device. Associating a short phrase describing the functionality of a button combination is one way to memorize what each combination does. Allowing the user to directly speak this short phrase removes the step of associating with a button, thus making speech more intuitive than buttons.

However, there still is the problem of knowing or learning what commands are available for a given environment in a given state. This is often referred to as the "habitability" problem. The richness of the English language allows a speaker to convey the exact same meaning by using completely different words. It is impossible to prepare actions for every possible human phrase that a user may utter during a session. The ability to make the user aware of what commands are supported in a given situation is presently a very large unsolved problem.

Today's technology still often requires slow, deliberate speech (i.e., a "Star Trek" voice). This is changing as more research and commercial speech recognition groups perfect their software and hardware. The goal in many peoples' minds is to achieve the ubiquitous voice-activated computer in Star Trek.

We have used a commercial system as a voice recognition tool on the Workbench. The system has worked well in terms of understanding different accents; foreign speakers for whom English is not a native language have been understood. We have limited the number of commands, thus limiting the user's learning curve. However, the fundamental limitation of voice recognition remains. Users still need to whisper to us "well, what do I say," since a user can't know the precise phraseology required. Progress toward natural language recognition are steps toward removing this limitation.

3.3 Simulated "Laser" Pointer ("Wand")

Our third interface method is a simulated laser pointer (wand). To create this, we modified a PC flight stick and programmed a virtual laser beam to appear to emanate from the wand. The wand's motion is detected by the addition of an internal magnetic tracker and the position of the laser is adjusted accordingly. The successful intersection of an object with the "laser" causes a bounding sphere to appear around the object.

Movement of an intersected object is enabled by pulling the wand trigger. The object is "grabbed" by the laser beam and may be moved to any location on the Workbench. Releasing the trigger caused the object to drop and fall to the surface. While the object is grabbed, it may be rotated by pushing the leftmost button and it may be tracked in or out (with alternate button pushes) by pushing the rightmost button. The latter is convenient for moving objects long distances.

The wand is also used for applications with terrain to provide a convenient and intuitive method for object scaling, rotation, and translation. Button actions combined with user hand movements permit zoom, translation, and rotation. Once the correct button combinations are pressed, moving the hand up/down generates a zoom. Moving the hand left/right or forward/back generates a pan and rotating the wand generates a rotation of the terrain. In addition to object intersection and movement and terrain motion, the wand may be employed as a query device.

The wand was designed to simplify the use of the Workbench. All of the above functions are enabled by movement of the wand or combinations of trigger pulls and button pushes involving only one trigger and two buttons. However, the wand does not fulfill the long-term goal of VR of fully natural interaction. In addition, the wand requires the user to learn a sequence of unnatural interaction techniques: combinations of buttons and triggers are required for each interaction. This is reasonable when the number of interactions remains small. However, having to insert into the virtual environment a menu of required button/trigger interactions would not, in our view, be an effective interactive method.

3.4 Conclusions about Interactive Methods

We have found all three interactive methods discussed above to be effective, although each has limitations. The next sections discuss two of our Workbench applications. For the first we use a combination of glove and voice, while the second uses the wand. We plan to perform user evaluation studies to determine which combinations of interfaces produce the most effective interactions.

4.0 A Medical Application

The first application we developed on the Workbench as an early proof of concept was to display a human skeleton on the Workbench and investigate interactive methods for manipulating body parts. We selected this because of the very natural paradigm involved. A doctor standing over a patient on an examining (or operating) table knows the procedures he will undertake. Even non-practical demonstrations, such as removing an organ, examining it, and replacing it, are in some sense "natural" whereby it is clear what needs to be done. This application uses:

- direct manipulation
- simple pinch gesture recognition
- speech recognition
- tracked stereographic projection

The model used for this application is a commercially purchased human adult skeleton with many of the major internal organs. An articulated glove model was used as an avatar for the user. The purpose of this application was to experiment with direct manipulation, simple pinch gesture recognition, and speech recognition. Head-tracked, off-axis stereo projection was used to pr ovide the user with the best possible perspective into the virtual environment. Figure 2 shows the medical application on the Workbench.

4.1 Interaction for the Medical Application

A user had two methods of interacting with the model. The first was using a direct manipulation method; the user literally translated his hand into the projected skeleton model. The intersection of the hot spot of the glove avatar with a bone group or organ invoked a highlighting callback. This indicated that the user had "selected" that bone group or organ. The completion of a pinching gesture when a bone group or organ was highlighted caused the selected item to become attached to the glove avatar. In effect, the user had grabbed that item. The user can then manipulate the item as if he was really holding it in his hand.

Direct manipulation worked well for large bone groups and coarse movements of selected items. However, there were problems selecting small groups due to the precision movements required that were not possible with the tracker system and in the real world highly dependent on tactile feedback. This is also true for replacing bone groups. Without modeling collisions with other bone groups and many other aspects of the simulation, a user could not place the bone groups back exactly where they belonged. Manipulation of the entire model also became an issue. The interface could have been constructed using keyboard key sequences, additional pinch gestures, or complicated combinations of both solutions. However, we felt that this would increase the burden on the user. We wanted to make the interface as intuitive as possible. Thus, we experimented with speech recognition to address some of these concerns.

The second method of interaction was speech recognition. We used the commercially available HARK system from BBN. This system is a user independent system that does not require any learning. It requires a pre-defined grammar, and thus has limited recognition capabilities. For this application, our grammar consisted of very simple and short commands.

The user could select (highlight) any bone group or organ that he could name (e.g. "select clavicle"). This was one way for a user to practice his anatomy. The user could also "grab" an already selected (highlighted) item or could directly request a specific bone group or organ to be grabbed (i.e. "grab heart"). The named item would animate up to the glove avatar. Using speech recognition, a user can grab any bone group or organ on or off screen, regardless of how difficult it may have been to select or grab that item using direct manipulation.

Finally, the user had very basic manipulation control over the entire human model. The user could "rotate" the model counter-clockwise, and he could "scroll" the model left, right, up, and down. These operations could only be done through the speech recognition system.

While primitive, this proof of concept provided a starting point for conversations with medical professionals and other researchers. Most of the medical professionals that viewed this application immediately saw the potential for educational use. However, they saw the system to be even more applicable to medical procedure visualization and planning. They would like to have the ability to visualize real patient 3-D x-ray, CAT, MRI, or other data sets in a near real time manner. This would allow a doctor to visualize and plan a medical procedure with the actual anatomy of the patient.

5.0 Situational Awareness Using the Workbench

In this section we describe a situational awareness Workbench system that was fielded in March, 1997 [Rosenblum et al., 1997].

5.1 Overall Requirements

Our task is to provide situational awareness for the complex logistical task of directing the movement of U.S. Marines and materiel over rugged terrain, day and night, in uncertain weather conditions. This difficulty is multiplied by the well-known dangers of amphibious assault, long considered the most difficult problem in warfare.

Even with the advent of computers and sophisticated decision-making software in Marine Corps Combat Operation Centers (COC), command and control are predominantly undertaken with paper maps and acetate overlays. This is a cumbersome, time consuming process. In addition, detailed maps and overlays can take several hours to print and distribute. There currently exists no overall picture of the battlespace that provides a commander with a dynamic range of resolution sufficient to track units ranging from aircraft carriers to six-Marine fire teams. Furthermore, a mechanism is needed to deliver information, on demand, concerning the status of any unit of interest (fuel supply, ammunition, casualties, etc.). The resolution and bandwidth requirements to deliver this "big picture" in 3D is beyond the capabilities of the PC's and low-end work stations typically found in a COC. The Workbench is one item being demonstrated for possible use in an Enhanced COC (ECOC). The goal of this preliminary demonstration is to show the Workbench's capability to represent a large area terrain on the Workbench at a resolution comparable to maps used in the field and to utilize a selection of the interactive techniques discussed above to manipulate icons representing forces and objects on that terrain. Figure 3 shows a mission planning application on the Workbench, while Figure 4 shows several Marines being trained in using the system discussed in this section.

5.2 Terrain and Texture

Reasonable 3D terrain resolution of an area the size of the training base (a 62 x 72 Km area) requires a minimum of 20,000 vertices for high resolution. Complicating the construction of the terrain was the requirement that a virtual "ocean" exist outside a road network bordering the training area. To this end it made sense to utilize a commercial modeling and terrain package that could both automatically construct terrain from raw Defense Mapping Agency (DMA) data and provide the tools to create a reasonably realistic ocean. DMA data used was a height field on a 100 m. grid for the area of interest.

Commercial software was used to read DMA data directly from CD, select a usable resolution, and employ Delauney triangulation to calculate vertices and place them efficiently. The construction of the ocean required that the texture map (see below) be applied to the terrain so that the borderline road network could be seen. Vertices lying outside the desired border could be selected in groups and their elevation decreased to zero. It was important to select only vertices that were not part of an edge that crossed into the land area. Otherwise, the terrain contours of the exercise area would have been distorted. Thus, a tradeoff was made that left some ocean vertices at an elevation of greater than zero. These were concealed using an alpha channel in the terrain texture.

Even more demanding are the requirements for texture-map resolution because, in addition to aerial and satellite photos, the terrain must be textured with line-drawing maps with a geographic resolution approaching 1:250,000. This requires a minimum of an 8kx8k pixel image that will occupy 125 MB of storage in raw format at the 16 bit (r-g-b-a = 5-5-5-1 bits) default color resolution for images in Iris Performer. Reduction of line drawings to more manageable sizes is impracticable because of the unavoidable loss of contours, grids, and text legibility (although a 2Kx2K map was used for ocean creation). This is beyond the capacity of the maximum texture memory for an Onyx (64 MB). Thus, we used a clip mapping approach to terrain texturing. Clip mapping hardware is standard on the Onyx Infinite Reality.

A large map image (8kx8k) of the exercise area was extracted from a DMA Compressed ARC Digitized Raster Graphics CD. This was too large to work with directly and our "ocean" had to be drawn on a 2Kx2K copy using Adobe Photoshop. The image was then scaled to 8kx8k and the monochrome ocean area was selectively copied and ultimately composited over the original 8kx8k image of our map using the San Diego Supercomputer Center's Imaging Tools.

An alpha channel was added to render the ocean area transparent. Modulation of the texture over the terrain resulted in the transparency of the polygons underlying the ocean. To create the clip-map texture, the final image was cut into a pyramid of 1Kx1K tiles; for example, 64 tiles from the 8Kx8K image, 16 tiles from a 4Kx4K image, etc., using various Silicon Graphics imaging tools. The clip mapping hardware in an Onyx/Infinite Reality Engine manages the paging of the 1Kx1K tiles between disk, RAM, and texture memory.

5.3 Models

Approximately half of the military models used were of commercial origin and were, typically, models of complex equipment such as tanks, ships, helicopters, etc. The simpler models were constructed at our laboratory. Large units (battalion size or larger) were represented by flags bearing the name and seal associated with that unit. Smaller units (platoons, squads, and fire teams, for example) were represented by simple cubes textured on all sides with standard military symbols such as an 'X' for infantry and a sideways 'E' for engineers. These are easily recognizable by the users.

Placement of units on the terrain may be achieved in two ways. The first uses a simple token scheme in a designated status file. Each token is followed by data such as unit name, lat/lon, altitude, etc. The file is read automatically by the program when a time change is detected. An output file of the same format may be saved at the discretion of the user. The second method uses on-the-fly electronic messaging. A separate application, initiated manually, downloads the email containing status updates. The new icons are introduced upon receipt of the email. The user maintains control of the updates, because they are delivered on demand. Figures 5 and 6 illustrate the terrain and models on the Workbench surface, displayable in either 3D or 3D stereo.

5.4 Interaction

An overall view of the battle space, including hundreds of operational units, is useful in itself for command decisions; however, more detailed information is needed to prosecute an action. To enable interrogation of icons and presentation of their underlying data, we modified a PC flight stick and programmed a virtual laser beam to appear to emanate from the flight stick.

The motion of the flight stick detected by the addition of an internal tracker and the position of the laser is adjusted accordingly. The intersection of an icon with the "laser" causes a bounding sphere to appear around the icon (indicating successful intersection) and a heads-up display (HUD) to appear on the right side of the workbench surface. The HUD displays all the information discussed above and automatically disappears five seconds after a unit is deselected or immediately when a new unit is chosen.

User controlled movement of an intersected icon is enabled by pulling the flight stick trigger. The object is "grabbed" by the laser beam and may be moved to any location on the terrain. Releasing the trigger causes the icon to drop and fall to the terrain surface. Aircraft are encoded to remain at whatever altitude they are released. While the icon is grabbed, it may be rotated by rotating the wand about the laser's axis and it may be tracked in or out (again with alternate button pushes) by pushing the rightmost button. The latter is convenient for moving icons long distances across the terrain.

The flight stick provides a convenient and intuitive method for scaling and translating the terrain. The following actions are always enabled with the trigger pulled in, the terrain intersected, and a button pushed. Pressing the leftmost button and raising the flight stick causes the terrain to uniformly scale up. Moving the flight stick left or right and forward or back, with the same button pressed, moves the terrain in the same direction. Pressing the rightmost button and rotating the flight stick around the Z-axis caused the terrain to rotate in the same direction around the Z-axis. Rotating the flight stick around the X-axis, with the same button pressed causes the terrain to rotate around the X-axis (change pitch).

In addition to icon intersection and movement and terrain motion, the flight stick may be employed as a measuring device. Whenever the laser intersects the terrain, a small HUD, in the lower right corner of the workbench, appears and displays the coordinates (lat./lon. and UTM) and the elevation (above sea level) at the point of intersection.

Distances and headings are measured by intersecting the first point of interest (or icon) with the laser and pressing only the left-most button. The second point is then intersected and the left-most button pressed again. A HUD appears along the lower edge of the screen and displays the distance, heading, and elevation change between two points or a series of points. Pressing the rightmost button resets the measurements and causes the HUD to disappear.

The flight stick concept was designed to simplify the use of the workbench. Indeed, all the above functions are enabled by movement of the flight stick or combinations of trigger pulls and button pushes involving only one trigger and two buttons. However, as the functionality of the application increases, so will the difficulty of providing simple and intuitive interactions. We are currently planning to perform evaluation testing of interface methods. The results of these evaluations will drive future interface development efforts.

6.0 Networking

An ongoing challenge for VR is to integrate VR with networking to facilitate remote collaboration in problems ranging from manufacturing through modeling and simulation. This issue can be subdivided into two classes. Some applications require complex interactions among a limited number of participants, while others, such as military simulations, require servicing thousands of players. Large, multi-user virtual environments must keep each entity aware of other's actions. This

places considerable demands on the workstation I/O, network bandwidth, and the underlying architecture. One approach to this challenge, developed at the Naval Postgraduate School, is NPSNET [Macedonia et al., 1994]. NPSNET is a large scale software package designed for networking that is capable of simulating articulated humans and ground and air vehicles in the DIS networked virtual environment of 250-300 players. NPSNET is the first 3D virtual environment to make effective use of the multicast backbone of Internet in order to avoid direct connections between all sites. It also makes extensive use of dead-reckoning to predict object position and reduce visual latency in low-bandwidth situations. The software architecture logically partitions a virtual environment by associating spatial, temporal, and functional classes with network multicast groups.

We have begun an investigation into networked Workbenches, jointly with the Graphics, Visualization, and Usability Center at the Georgia Institute of Technology. For the Workbench, the problem is not dealing with a large number of users. Rather, the Workbench emphasizes fine-grained interactions. The detailed interaction raises many interesting issues in human perception, interaction, and collaboration. Questions of how to partition the usable workspace, of what operations can and cannot you perform (remotely) on my Workbench, of how two remote users can share a common object, and similar issues will be the topic of future investigations. We have recently completed a first demonstration of networked Workbenches. The workbenches are connected by an ATM network and each viewer sees correct perspective. Issues of how joint collaboration should be performed ("ownership") are under investigation.

7.0 Conclusions

The Virtual Reality Responsive Workbench is fundamentally different from previous VR systems in that it emphasizes fine-grained interaction rather than navigation through immersed space (or, in some systems, just 3D viewing with 2D interaction techniques). Only four years old, the Responsive Workbench is rapidly being accepted as a major VR paradigm. It has transitioned from NRL to VR research universities, to commercial development of the hardware, and to implemented, utilized systems such as the situational awareness application discussed in this article. A number of research and development issues are first being examined. These include graphical representations on the workbench, interface issues, topics in perception and evaluation, and effective systems for networking the Workbench. Hardware improvements are also important, particularly we look forward to the time when the projector can be replaced by flat panel displays. We anticipate a lot of activity in Workbench development over the next five years by many organizations. We see Workbenches moving "out of the lab" and into command centers, engineering design centers, medical training centers, and to other end users with similar requirements.

Acknowledgments

The authors thank others who have contributed to these projects: Rob King, Brad Colbert, and Chris Scannell as well as graduate student summer interns Greg Newton (Georgia Tech), Jim Van Verth (Univ. of North Carolina), and Ken Gordon (SUNY Stony Brook). This work was sponsored by the U.S. Office of Naval Research, the Marine Corps Warfighting Laboratory, and the Defense Advanced Projects Research Agency.

References:

[Cruz-Neira et al., 1993] Cruz-Neira, C., D. Sandin, T. DeFanti, "Surround-screen, projection-based virtual reality: The design and implementation of the CAVE," *Computer Graphics* (Proc. Siggraph 93), 135-142, July 1993

[Durlach and Mavor, 1995] Durlach, N.I., and A.S. Mavor (Eds.), *Virtual Reality: Scientific and Technological Challenges*, National Academy Press, 1995.

[Earnshaw and Rosenblum, 1995] Earnshaw, R.A., and L.J. Rosenblum, "Virtual reality, visualization, and their application," *Proc. Interface 95*.

[Kreuger and Froehlich, 1994] Kreuger, W. and B. Froehlich, "The Responsive Workbench," IEEE Computer Graphics and Applications, Vol. 14, No. 3, May, 1994, pp. 12-15.

[Kreuger et al., 1995] Kreuger, W., C. Bohn, B. Froehlich, H. Schueth, W. Strauss, and G. Wesche, "The Responsive Workbench: A Virtual Work Environment", IEEE Computer, Vol. 28, No. 7, July, 1995, pp. 42-48.

[Macedonia et al., 1994] Macedonia, M.R., M.J. Zyda, D.R. Pratt, P.T. Barham, and S. Zeswitz, "NPSNET: A network software architecture for large scale virtual environments," *Presence*, Vol. 3, No. 4, Fall 1994.

[Obeysekare et al., 1996] Obeysekare, U., C. Williams, J. Durbin, L. Rosenblum, R. Rosenberg, F. Grinstein, R. Ramamurthi, A Landsberg, and W. Sandberg, "Virtual Workbench - A non-immersive virtual environment for visualizing and interacting with 3D objects for scientific visualization," *Proc. Visualization 96*, Oct. 1996, pp. 345-350.

[Rosenblum, Bryson, and Feiner, 1995]Rosenblum, L., S. Bryson, and S. Feiner, "Virtual reality unbound," *IEEE CG&A*, Vol. 15, No. 5, Sept. 1995, pp. 19-21.

[Rosenblum et al., 1996] Rosenblum, L. J., J. Durbin, L. Sibert, D. Tate, J. Templeman, U. Obeysekare, J. Agrawaal, D. Fasulo, T. Myers, G. Newton, A. Shalev, "Shipboard VR: from damage control to design", *IEEE Computer Graphics and Applications*, Vol. 15, No. 6, Nov. 1996.

[Rosenblum and Cross, 1997] Rosenblum and R.A. Cross, "Challenges in Virtual Reality," to appear in Visualization and Modelling, Academic Press 1997.

[Rosenblum et al., 1997] Rosenblum, L.J., J. Durbin, R. Doyle, and D. Tate, “Situational Awareness Using the VR Responsive Workbench,” *IEEE Computer Graphics and Applications*, Vol. 16, No. 4, July, 1997.